

## MapReduce 读后感

天气预报、城市外来人口统计等很多现实问题都需要处理以 TB 计算的大量数据集，用一台高性能的电脑处理不仅成本高，出错率高及速度慢这些问题都令人头疼。为了解决这个问题，mapreduce 随应到来，这种编程模型利用分布式计算思想，把大量数据拆分给多个低性能电脑，处理后的结果汇总输出。我们只需要输入需要定义任务、去除错误的数据，静等结果。这种处理方法的优点在于：低性能的电脑省去了昂贵的成本，多台电脑的处理也增加了容错率，并行处理大大减少了处理时间。

MapReduce 架构的程序能够在大量的普通配置的计算机上实现并行化处理。这个系统在运行时只关心：如何分割输入数据，在大量计算机组成的集群上的调度，集群中计算机的错误处理，管理集群中计算机之间必要的通信。采用 MapReduce 架构可以使那些没有并行计算和分布式处理系统开发经验的程序员有效利用分布式系统的丰富资源。

Mapreduce 模型分 map、reduce 两部分。根据谷歌三篇大数据论文，MapReduce 编程模型的原理是：用户自定义的 Map 函数接受一个输入的 key/value pair 值，然后产生一个中间 key/value pair 值的集合。MapReduce 库把所有具有相同中间 key 值 I 的中间 value 值集合在一起后传递给 reduce 函数。用一个例子详细解释 mapreduce 思想。要求数出 1000 张纸牌中有多少张黑桃，100 个人每人随机发放 10 张，每个人数出自己拿到的牌中有多少张黑桃，这是 map 函数实现的，reduce 函数的作用是将 100 人数的张数加在一起输出。Key 是黑桃，value 是黑桃的张数。

在了解了理论知识后，如何进行实际操作呢？在处理数据的多台电脑中有一个特殊的存在——master。Master 统领大局，分配任务给剩余的计算机——worker。worker 处理分配的数据，找出 key 和 value 传递给 Map 函数生成并输出的中间 key/value 并缓存在内存中。Key 和 value 在磁盘上的位置将被回传给 master，master 再把这些位置发给 worker。Worker 从相应位置上读取这些缓存数据。当 Reduce worker 读取了所有的中间数据后，通过对 key 进行排序使得具有相同 key 值的数据聚合在一起。Worker 将中间的 key 和 value 集合传递给 Reduce 函数统计相加。因为参与运算的都是低性能的计算机，master 可能会罢工，这时候就有全体 worker 投票选出一个新的 master。为防止 worker 罢工，有一个程序让 worker 每隔一定时间向 master 发送信号，当 master 接收不到信号时，master 会将任务分配给其他的 worker。当 worker 故障时，由于已经完成的 Map 任务的输出存储在这台机器上，Map 任务的输出已不可访问了，因此必须重新执行。而已经完成的 Reduce 任务的输出存储在全局文件系统上，因此不需要再次执行。这种机制有效降低了出错率。

作为一名大三的学生，想要完全理解谷歌的这篇论文还需要长期的积累，现在最重要的是通过这篇论文学习大数据的思想，放宽视野、立足世界。就像老师说的那样，通过一台可以联网的设备可以联系世界各地的电脑，共同解决某一问题；充分利用学校多个专业的资源，发现大数据在不同领域的作用。以上是老师所讲的内容结合自己的理解，不足之处还望指正。